In the claims:

Presented below are the claims, as amended, with changes entered and unmarked. For the Examiner's convenience, all pending claims are presented herein. Those claims that remain unchanged by this amendment are prefixed with "(Unchanged)". Please cancel without prejudice claims 3, 14, and 19. Please add claims 38-41. Please amend the claims as follows:

---

1. (Amended) A computer system comprising:

a cache memory having a plurality of cache lines each of which stores data;

a storage area to store a data operand; and

an execution unit coupled to said storage area to operate on data elements in said data operand containing a portion of a starting address to invalidate data in a predetermined portion of the plurality of cache lines in response to receiving a single instruction of a processor instruction set.

---

2. (Unchanged) The computer system of Claim 1, wherein the data operand is a register location.

---

4. (Amended) The computer system of Claim 1, wherein the portion of the starting address includes a plurality of most significant bits of the starting address.

---

5. (Unchanged) The computer system of Claim 4, wherein execution unit shifts the data elements by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

6. (Unchanged) The computer system of Claim 1, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

7. (Amended) A computer system comprising:

a first storage area to store data;

a cache memory having a plurality of cache lines each of which stores data;

a second storage area to store a data operand containing a portion of an address; and

an execution unit coupled to said first storage area, said second storage area, and said cache memory, said execution unit to operate on the portion of an address in said data operand to copy data from a predetermined portion of the plurality of cache lines in the cache memory to the first storage area, in response to receiving a single instruction of a processor instruction set.

8. (Unchanged) The computer system of claim 7, wherein the data operand is a register location.

9. (Unchanged) The computer system of claim 8, wherein the register location contains a plurality of most significant bits of a starting address of the cache line in which data is to be copied.

10. (Amended) The computer system of claim 9, wherein execution unit shifts the portion of an address by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be copied.

11. (Unchanged) The computer system of Claim 7, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

12. (Unchanged) The computer system of Claim 7, wherein the execution unit further invalidates data in the predetermined portion of the plurality of cache lines in response to receiving the single instruction, upon copying the data to the first storage area.

13. (Amended) A processor comprising:

a decoder configured to decode instructions, and

a circuit coupled to said decoder, said circuit in response to a single decoded instruction of a processor instruction set being configured to:

read a portion of an address located in a register specified in the decoded instruction to obtain a starting address of a predetermined area of a cache memory on which the instruction will be performed:

invalidate in the predetermined area of cache memory.

15. (Amended) The processor of Claim 13, where in the portion of an address includes a plurality of most significant bits of the starting address.

16. (Amended) The processor of Claim 15, wherein the circuit shifts the portion of an address by predetermined number of bits positions to obtain the starting address of the cache line in which data is to be invalidated.

17. (Unchanged) The processor of Claim 13, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

18. (Amended) A processor comprising:

a decoder configured to decode instructions, and

a circuit coupled to said decoder, said circuit in response to a single decoded instruction of a processor instruction set being configured to:

read a portion of an address located in a register specified in the decoded instruction to obtain a starting address of a predetermined area of a cache memory on which the instruction will be performed;

copy data in the predetermined area of cache memory;

store the coupled data in storage area separate from the cache memory.

20. (Amended) The processor of Claim 18, wherein the portion of an address includes a plurality of most significant bits of the starting address.

21. (Amended) The processor of Claim 20, wherein the circuit shifts the portion of an address by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be copied.

22. (Unchanged) The processor of Claim 20, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

23. (Unchanged) The processor of Claim 20, wherein said circuit further invalidates the data in the predetermined portion of the plurality of cache lines in response to receiving the single instruction, upon copying the data to the storage area.

24. (Amended) A computer-implemented method, comprising:

a) decoding a single instruction of a processor instruction set;

b) in response to said decoding the single instruction, obtaining a portion of a starting address of a predetermined areas of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) completing execution of said single instruction by invalidating data in the predetermined area of cache memory.

25. (Unchanged) The method of Claim 24, wherein c) comprises setting an invalid bit corresponding to the predetermined area of cache memory.

26. (Amended) The method of Claim 24, wherein b) comprises:

shifting the portion of the starting address by the predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

27. (Amended) The method of Claim 26, wherein in the portion of the starting address contains a plurality of most significant bits of the starting address, and wherein in the predetermined number of bit positions represent the number of least significant bits of the starting address

28. (Unchanged) The method of Claim 24, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

29. (Amended) A computer-implemented method, comprising:

a) decoding a single instruction of a processor instruction set;

b) in response to said decoding the single instruction, obtaining a potion of a starting address of a predetermined area of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) completing execution of said single instruction by copying data in the predetermined area of cache memory and storing the copied data in a storage area separate from the cache memory.

30. (Unchanged) The method of Claim 29, wherein c) comprises setting an invalid bit corresponding to the predetermined area of cache memory.

31. (Amended) The method of Claim 29, wherein b) comprises:

shifting the portion of the starting address by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

32. (Amended) The method of Claim 31, wherein in the portion of the starting address contains a plurality of most significant bits of the starting address, and wherein in the

predetermined number of bit positions represent the number of least significant bits of the starting address.

33. (Unchanged) The method of Claim 29, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

34. (Unchanged) The method of Claim 29, further comprises:

d) invalidating the data in the predetermined portion of the plurality of cache lines in response to receiving the single instruction, upon copying the data to the storage area.

35. (Amended) A computer-readable apparatus, comprising:

a computer-readable medium that stores an instruction which when executed by a processor causes said processor to:

a) decode a single instruction of a processor instruction set;

b) in response to decoding the single instruction, obtain a portion of a starting address of a predetermined areas of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) complete execution of said single instruction by invalidating data in the predetermined area of cache memory.

36. (Amended) A computer-readable apparatus comprising:

a computer-readable medium that stores an instruction which when executed by a processor causes said processor to:

a) decode a single instruction of a processor instruction set;

b) in response to said decoding the single instruction, obtain a potion of a starting address of a predetermined area of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) complete execution of said single instruction by copying data in the predetermined area of cache memory and storing the copied data in a storage area separate from the cache memory.

37. (Unchanged) The apparatus of Claim 36, wherein the instruction further causes the processor to:

invalidate the data in the predetermined portion of the plurality of cache lines in response to receiving the instruction, upon copying the data to the storage area.

Please add the following new claims:

---

38. (New) A computer system comprising:

a cache memory having a plurality of cache lines each of which stores data;

a storage area to store a data operand; and

an execution unit coupled to said storage area to operate on data elements in said data operand identifying a user-definable linear or physical address identifying a predetermined portion of the plurality of cache lines to invalidate data in the predetermined portion of the plurality of cache lines in response to receiving a single cache control instruction of a processor instruction set, the single cache control instruction including a reference to the data operand.

39. (New) The computer system of Claim 38, wherein the data operand is a register location.

40. (New) The computer system of Claim 39, wherein execution unit shifts the data elements by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

41. (New) The computer system of Claim 38, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

AMENDMENT

Marked Version

In the specification:

Please replace the following paragraphs:

- Page 10, paragraph starting at line 1 and ending at line 7:

The registers 141 represent a storage [are] area on computer system 105 for storing information, such as control/status information, scalar and/or packed integer data, floating point data, etc. It is understood that one aspect of the invention is the described instruction set. According to this aspect of the invention, the storage area used for storing the data is not critical. The term data processing system is used herein to refer to any machine for processing data, including the computer system(s) described with reference to Figure 1.

- Page 10, paragraph starting at line 8 and ending at line 16:

Figure 2 illustrates one embodiment of the format of any one of the cache segment invalidate [instruction] instructions 162, the cache segment flush [instructions] instruction 164, and the cache segment flush and invalidate [instruction] instructions 166 provided in accordance with the present invention. For discussion purposes, the instructions 162, 164 and 166 will be referred to as the cache control instruction 160. The cache control instruction 160 comprises and operational code (OP CODE) 210 which identifies the operation of the cache control instruction 160 and an operand 212 which specifies the name of a register of memory location which holds a starting address of the data object that the instruction 160 will be operating on.

In the claims:

Presented below are the claims, as amended, with changes entered and marked. For the Examiner's convenience, all pending claims are presented herein. Those claims that remain unchanged by this amendment are prefixed with "(Unchanged)". Please cancel without prejudice claims 3, 14, and 19. Please add claims 38-41. Please amend the claims as follows:

1. (Amended) A computer system comprising:

a cache memory having a plurality of cache lines each of which stores data;

a storage area to store a data operand; and

an execution unit coupled to said storage area to operate on data elements in said data operand containing a portion of a starting address to invalidate data in a predetermined portion of the plurality of cache lines in response to receiving a single instruction of a processor instruction set.

2. (Unchanged) The computer system of Claim 1, wherein the data operand is a register location.

4. (Amended) The computer system of Claim [3] 1, wherein the portion of the starting address includes a plurality of most significant bits of the starting address.

5. (Unchanged) The computer system of Claim 4, wherein execution unit shifts the data elements by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

6. (Unchanged) The computer system of Claim 1, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

7. (Amended) A computer system comprising:

a first storage area to store data;

a cache memory having a plurality of cache lines each of which stores data;

a second storage area to store a data operand containing a portion of an address; and

an execution unit coupled to said first storage area, said second storage area, and said cache memory, said execution unit to operate on [data elements] the portion of an address in said data operand to copy data from a predetermined portion of the plurality of cache lines in the cache memory to the first storage area, in response to receiving a single instruction of a processor instruction set.

8. (Unchanged) The computer system of claim 7, wherein the data operand is a register location.

9. (Unchanged) The computer system of claim 8, wherein the register location contains a plurality of most significant bits of a starting address of the cache line in which data is to be copied.

10. (Amended) The computer system of claim 9, wherein execution unit shifts the [data elements] portion of an address by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be copied.

11. (Unchanged) The computer system of Claim 7, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

12. (Unchanged) The computer system of Claim 7, wherein the execution unit further invalidates data in the predetermined portion of the plurality of cache lines in response to receiving the single instruction, upon copying the data to the first storage area.

13. (Amended) A processor comprising:

a decoder configured to decode instructions, and

a circuit coupled to said decoder, said circuit in response to a single decoded instruction of a processor instruction set being configured to:

read a portion of an address located in a register specified in the decoded instruction of a processor instruction set to obtain a starting address of a predetermined area of a cache memory on which the instruction will be performed:

invalidate in the predetermined area of cache memory.

15. (Amended) The processor of Claim 13, where in the portion of [the starting] an address includes a plurality of most significant bits of the starting address.

16. (Amended) The processor of Claim 15, wherein the circuit shifts the [data elements] portion of an address by predetermined number of bits positions to obtain the starting address of the cache line in which data is to be invalidated.

17. (Unchanged) The processor of Claim 13, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

18. (Amended) A processor comprising:

a decoder configured to decode instructions, and

a circuit coupled to said decoder, said circuit in response to a single decoded instruction of a processor instruction set being configured to:

read a portion of an address located in a register specified in the decoded instruction to obtain a starting address of a predetermined area of a cache memory on which the instruction will be performed;

copy data in the predetermined area of cache memory;

store the coupled data in storage area separate from the cache memory.

20. (Amended) The processor of Claim 18, wherein the portion of [the starting] an address includes a plurality of most significant bits of the starting address.

21. (Amended) The processor of Claim 20, wherein the circuit shifts the [data elements] portion of an address by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be copied.

22. (Unchanged) The processor of Claim 20, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

23. (Unchanged) The processor of Claim 20, wherein said circuit further invalidates the data in the predetermined portion of the plurality of cache lines in response to receiving the single instruction, upon copying the data to the storage area.

24. (Amended) A computer-implemented method, comprising:

a) decoding a single instruction[:] of a processor instruction set;

b) in response to said [step of] decoding the single instruction, obtaining a portion of a starting address of a predetermined areas of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) completing execution of said single instruction by invalidating data in the predetermined area of cache memory.

25. (Unchanged) The method of Claim 24, wherein c) comprises setting an invalid bit corresponding to the predetermined area of cache memory.

26. (Amended) The method of Claim 24, wherein b) comprises:

[b.1) obtaining a portion of the starting address from a storage location specified in the decoded instruction;]

[b.2)] shifting the portion of the starting address by the predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

27. (Amended) The method of Claim 26, wherein in [b.1]] the portion of the starting address contains a plurality of most significant bits of the starting address, and wherein in [b.2)] the predetermined number of bit positions represent the number of least significant bits of the starting address

28. (Unchanged) The method of Claim 24, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

29. (Amended) A computer-implemented method, comprising:

a) decoding a single instruction of a processor instruction set;

b) in response to said [step of] decoding the single instruction, obtaining a potion of a starting address of a predetermined area of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) completing execution of said single instruction by copying data in the predetermined area of cache memory and storing the copied data in a storage area separate from the cache memory.

30. (Unchanged) The method of Claim 29, wherein c) comprises setting an invalid bit corresponding to the predetermined area of cache memory.

31. (Amended) The method of Claim 29, wherein b) comprises:

[b.1) obtaining a portion of the starting address from a storage location specified in the decoded instruction;]

[b.2)] shifting the portion of the starting address by a predetermined number of bit positions to obtain the starting address of the cache line in which data is to be invalidated.

32. (Amended) The method of Claim 31, wherein in [b.1]] the portion of the starting address contains a plurality of most significant bits of the starting address, and wherein in [b.2),] the predetermined number of bit positions represent the number of least significant bits of the starting address.

33. (Unchanged) The method of Claim 29, wherein the predetermined portion of the plurality of cache lines is a page in the cache memory.

34. (Unchanged) The method of Claim 29, further comprises:

d) invalidating the data in the predetermined portion of the plurality of cache lines in response to receiving the single instruction, upon copying the data to the storage area.

35. (Amended) A computer-readable apparatus, comprising:

a computer-readable medium that stores an instruction which when executed by a processor causes said processor to:

[obtain a starting address of a predetermined area of cache memory on which the instruction will be performed; and

invalidate data in the predetermined area of cache memory.]

a) decode a single instruction of a processor instruction set;

b) in response to decoding the single instruction, obtain a portion of a starting address of a predetermined areas of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) complete execution of said single instruction by invalidating data in the predetermined area of cache memory.

36. (Amended) A computer-readable apparatus comprising:

a computer-readable medium that stores an instruction which when executed by a processor causes said processor to:

[obtain a starting address of a predetermined area of a cache memory on which the instruction will be performed;

copy data from the predetermined area of cache memory; and

store the copied data in a storage area separate from the cache memory.]

a) decode a single instruction of a processor instruction set;

b) in response to said decoding the single instruction, obtain a potion of a starting address of a predetermined area of a cache memory on which the single instruction will be performed by reading a portion of an address contained in a storage location specified in the decoded instruction; and

c) complete execution of said single instruction by copying data in the predetermined area of cache memory and storing the copied data in a storage area separate from the cache memory.

37. (Unchanged) The apparatus of Claim 36, wherein the instruction further causes the processor to:

invalidate the data in the predetermined portion of the plurality of cache lines in response to receiving the instruction, upon copying the data to the storage area.

Please add the following new claims:

38. Claim 38 is New.

39. Claim 39 is New.

40. Claim 40 is New.

41. Claim 41 is New.